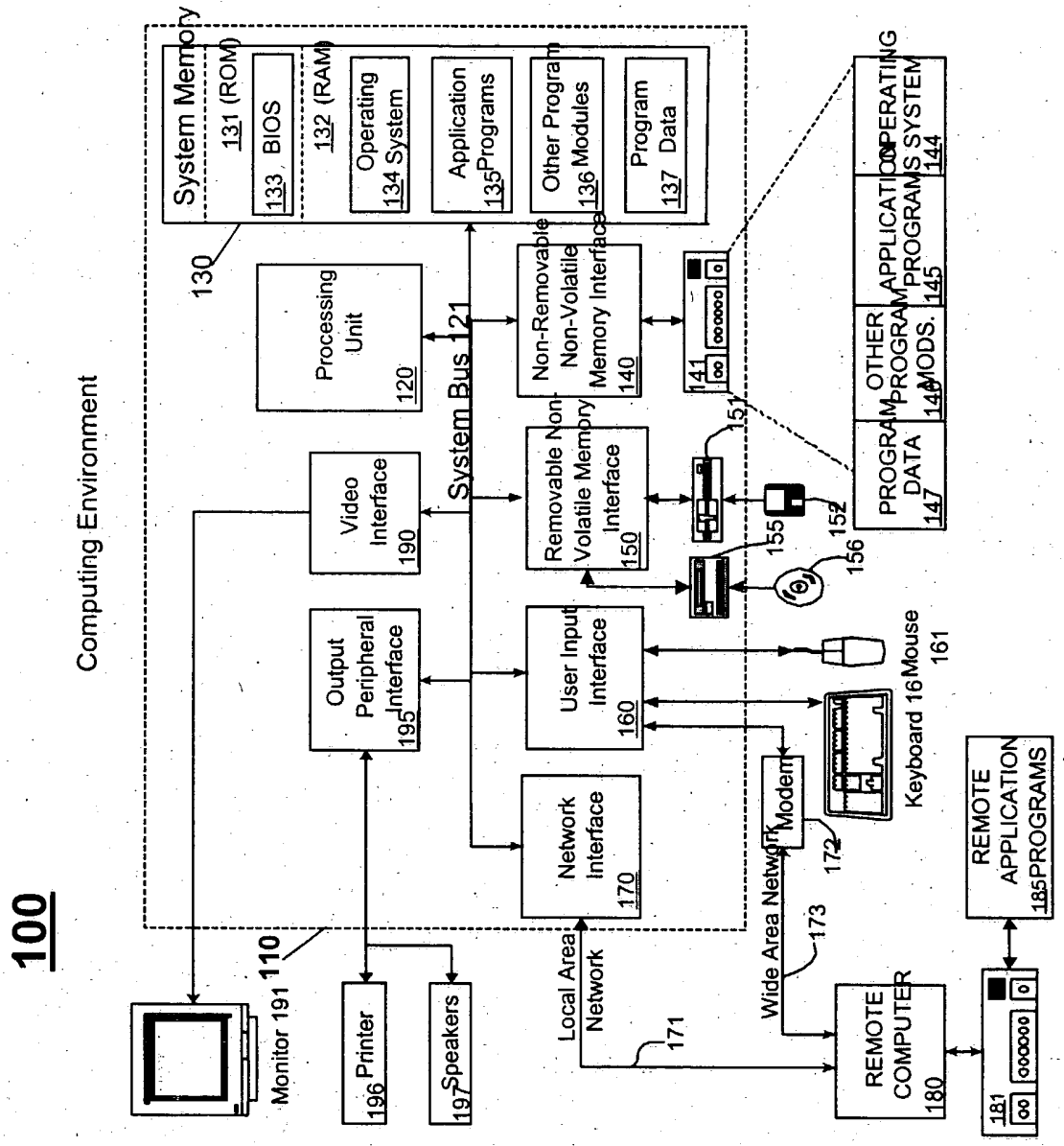

100
Computing Environment
System Memory
131 (ROM)
133 BIOS
132 (RAM)
Operating 134 System
Application 135 Programs
Other Program 136 Modules
Program 137 Data
130
Processing Unit 120
System Bus 121
Non-Removable Non-Volatile Memory Interface 140
141
OPERATING PROGRAMS SYSTEM 144
APPLICATION PROGRAMS 145
OTHER PROGRAM MODS. 146
PROGRAM DATA 147
Video Interface 190
Removable Non-Volatile Memory Interface 150
151
152
155
156
Output Peripheral Interface 195
User Input Interface 160
Mouse 161
Keyboard 162
Network Interface 170
Modem 172
Wide Area Network 173
Local Area Network 171
REMOTE APPLICATION PROGRAMS 185
REMOTE COMPUTER 180
181
Monitor 191
110
Printer 196
Speakers 197

FIGURE 1

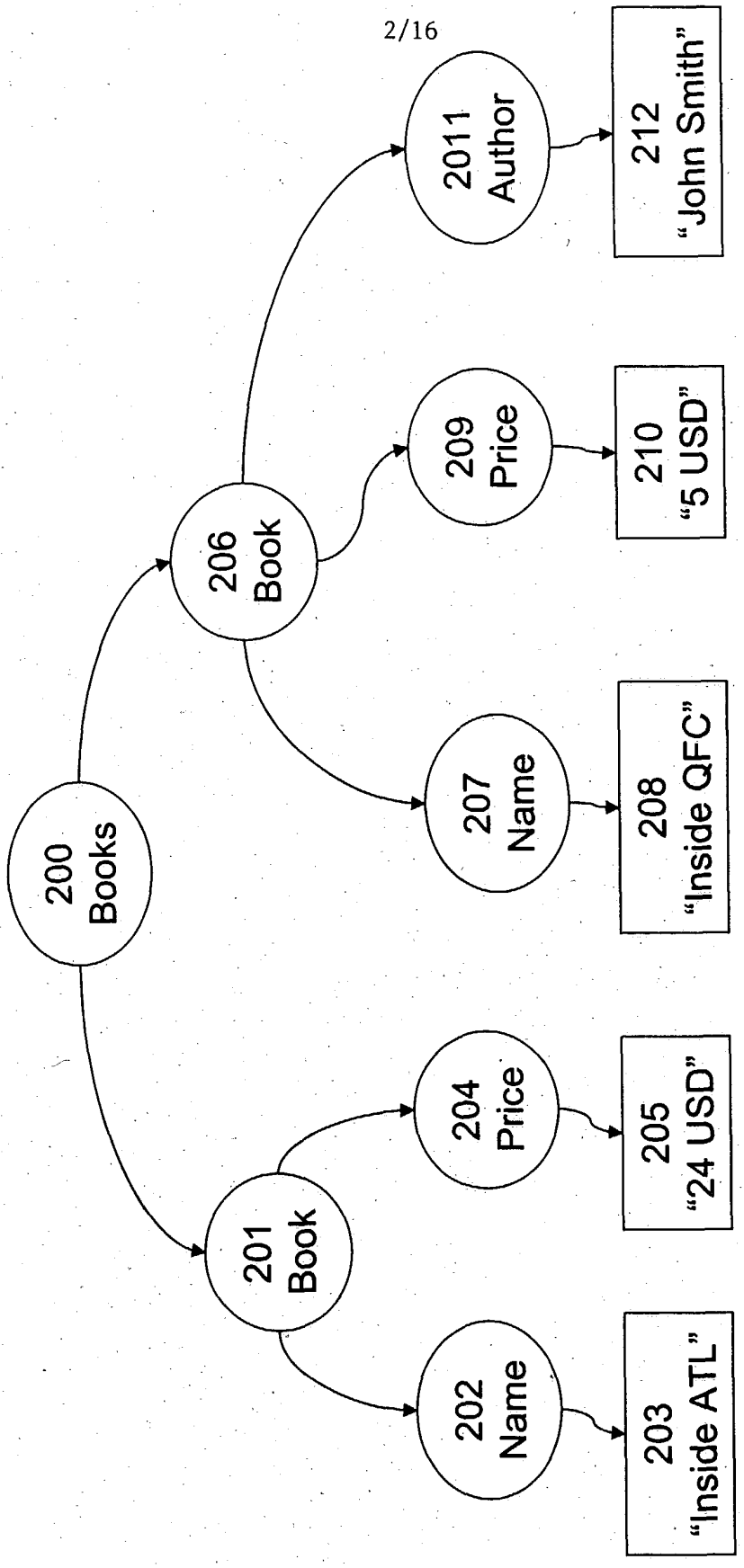

200 Books
206 Book
2011 Author
212 "John Smith"
209 Price
210 "5 USD"
207 Name
208 "Inside QFC"
201 Book
204 Price
205 "24 USD"
202 Name
203 "Inside ATL"

FIGURE 2A

```
- <books>
  - <book>
      <name>Inside ATL  </name>
      <price>24 USD</price>
    </book>
  - <book>
      <name>Inside QFC</name>
      <price>5 USD</price>
      <author>John Smith</author>
    </book>
  </books>
```

Serialized representation of hierarchical document of Figure 2A in XML format

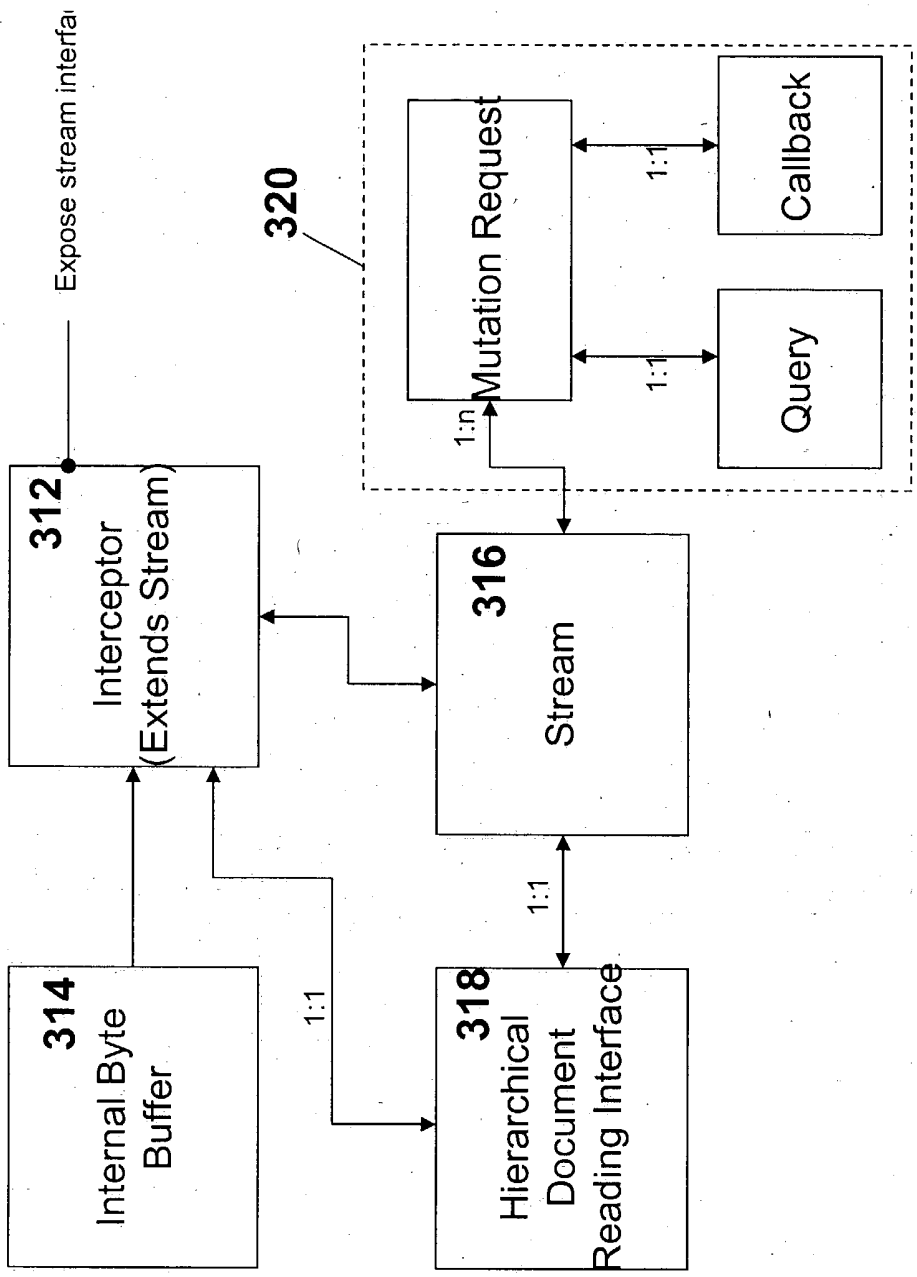
FIGURE 2B

00000000	3C	62	6F	6F	6B	73	3E	3C	62	6F	6F	6B	3E	3C	6E	61	<books><book><na
00000010	6D	65	3E	49	6E	73	69	64	65	20	41	54	4C	3E	3C	2F	me>Inside ATL </
00000020	6E	61	6D	65	3E	3C	70	72	69	63	65	3E	32	34	20	55	name><price>24 U
00000030	53	44	3C	2F	70	72	69	63	65	3E	3C	2F	62	6F	6F	6B	SD</price></book
00000040	3E	3C	62	6F	6F	6B	3E	3C	6E	61	6D	65	3E	49	6E	73	><book><name>Ins
00000050	69	64	65	20	51	46	43	3C	2F	6E	61	6D	65	3E	3C	70	ide QFC</name><p
00000060	72	69	63	65	3E	35	20	55	53	44	3C	2F	70	72	69	63	rice>5 USD</pric
00000070	65	3E	3C	61	75	74	68	6F	72	3E	4A	6F	68	6E	20	53	e><author>John S
00000080	6D	69	74	68	3C	2F	61	75	74	68	6F	72	3E	3C	2F	62	mith</author></b
00000090	6F	6F	6B	3E	3C	2F	62	6F	6F	6B	73	3E					ook></books>

FIGURE 2C

Token 1: Begin contain r node "books"
Token 2: Begin container node "book"
Token 3: Begin container node "name"
Token 4: Value node "Inside ATL"
Token 5: End container node "name"
Token 6: Begin container node "price"
Token 7: Value node "24 USD"
Token 8: End container node "price"
Token 9: End container node "book"
Token 10: Begin container node "book"
Token 11: Begin container node "name"
Token 12: Value node "Inside QFC"
Token 13: End container node "name"
Token 14: Begin container node "price"
Token 15: Value node "5 USD"
Token 16: End container node "price"
Token 17: Begin container node "author"
Token 18: Value node "John Smith"
Token 19: End container node "author"
Token 20: End container node "book"
Token 21: End container node "books"

FIGURE 2D



Mutating Interceptor Class Diagram

FIGURE 3A

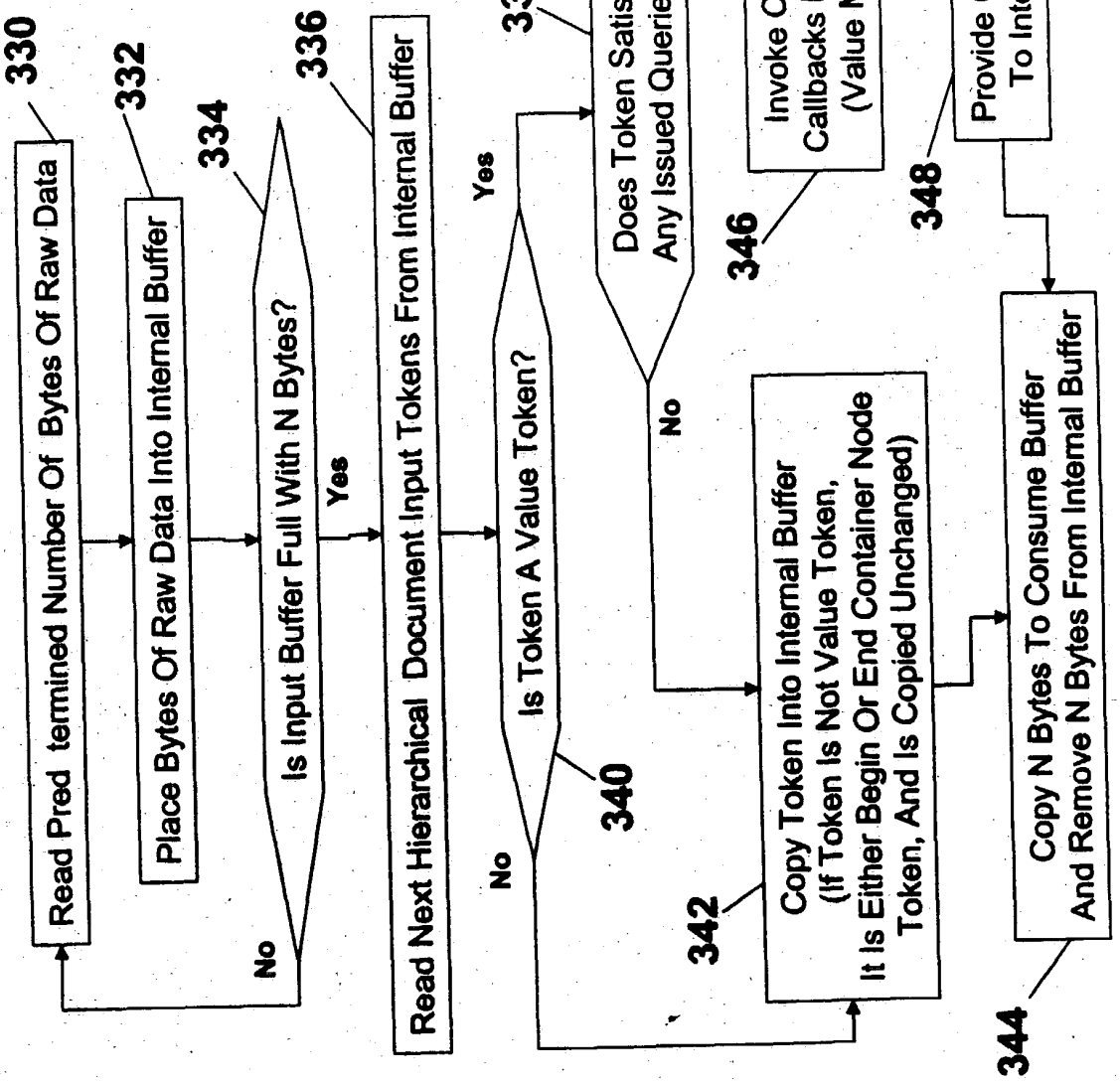
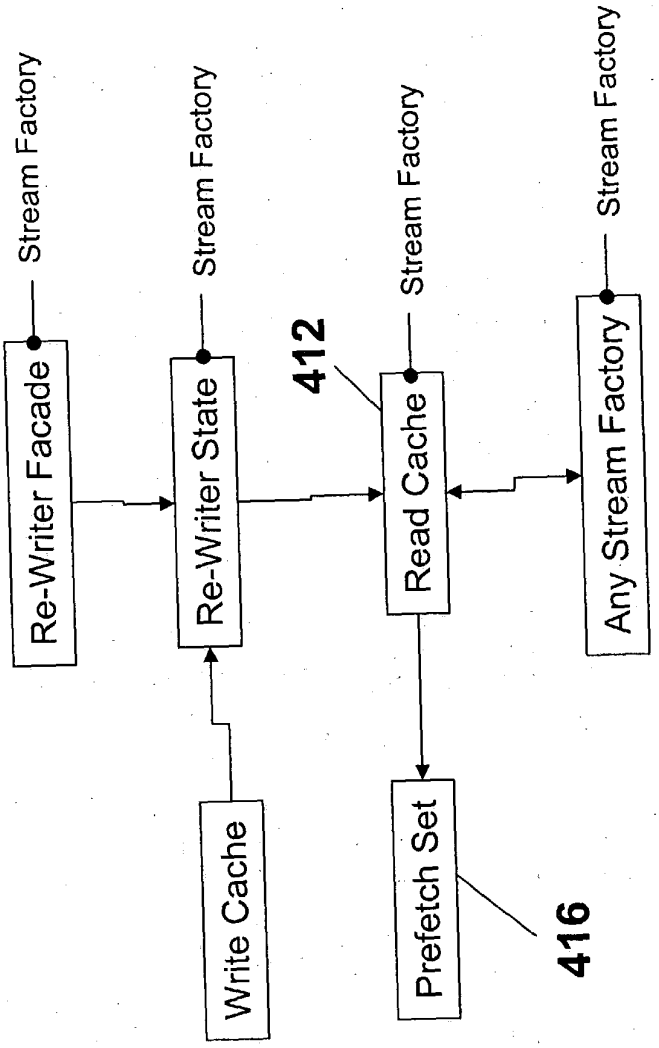

330
Read Pred termined Number Of Bytes Of Raw Data
332
Place Bytes Of Raw Data Into Internal Buffer
334
Is Input Buffer Full With N Bytes?
No
Yes
336
Read Next Hierarchical Document Input Tokens From Internal Buffer
338
Is Token A Value Token?
No
Yes
340
Does Token Satisfy Any Issued Queries?
No
Y s
342
Copy Token Into Internal Buffer (If Token Is Not Value Token, It Is Either Begin Or End Container Node Token, And Is Copied Unchanged)
346
Invoke Corresponding Callbacks Providing Value (Value May Change)
348
Provide Current Value To Internal Buffer
344
Copy N Bytes To Consume Buffer And Remove N Bytes From Internal Buffer

FIGURE 3B



Re-Writer Class Diagram

FIGURE 4

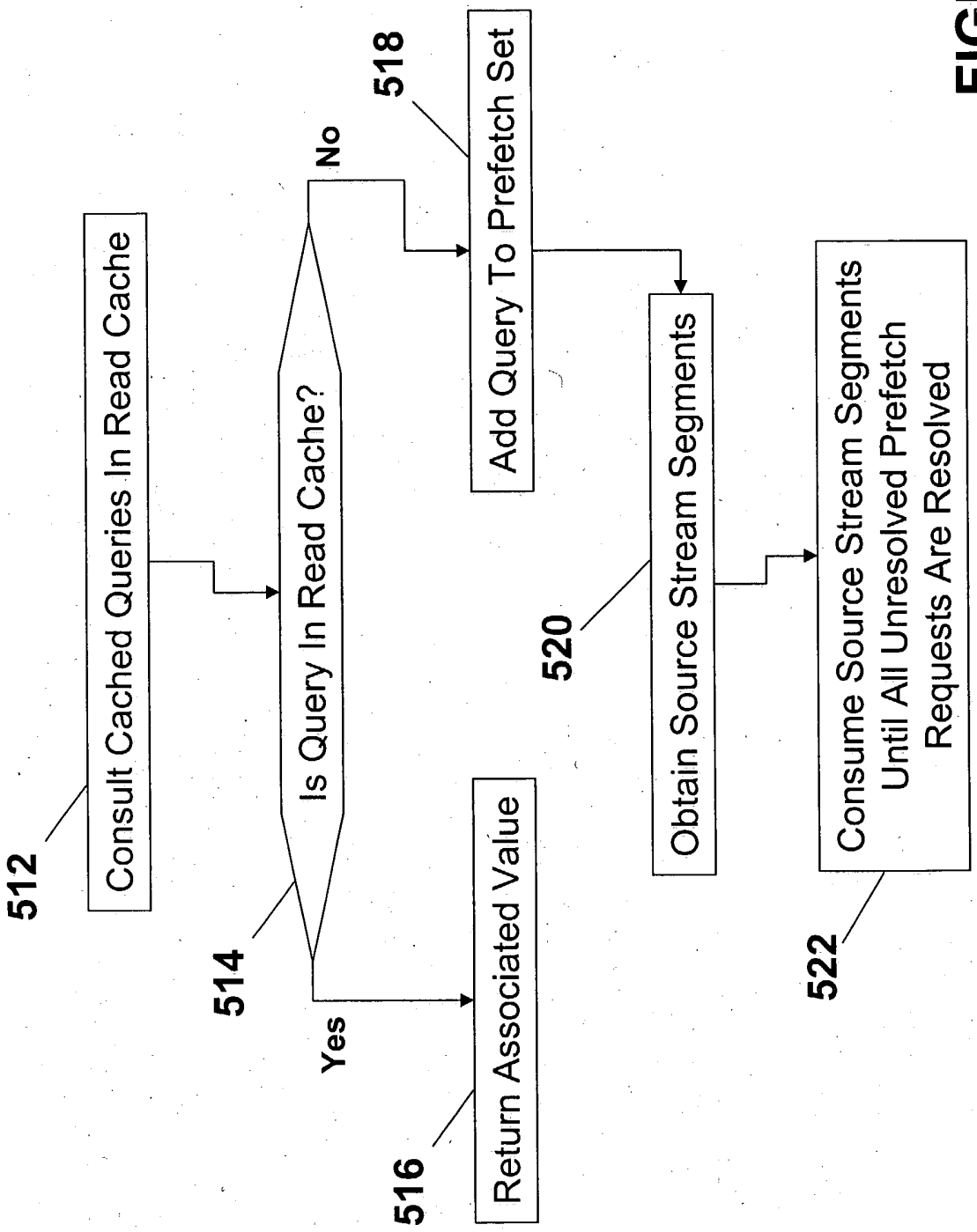

512
Consult Cached Queries In Read Cache
514
Is Query In Read Cache?
No
518
Add Query To Prefetch Set
Yes
516
Return Associated Value
520
Obtain Source Stream Segments
522
Consume Source Stream Segments Until All Unresolved Prefetch Requests Are Resolved

FIGURE 5

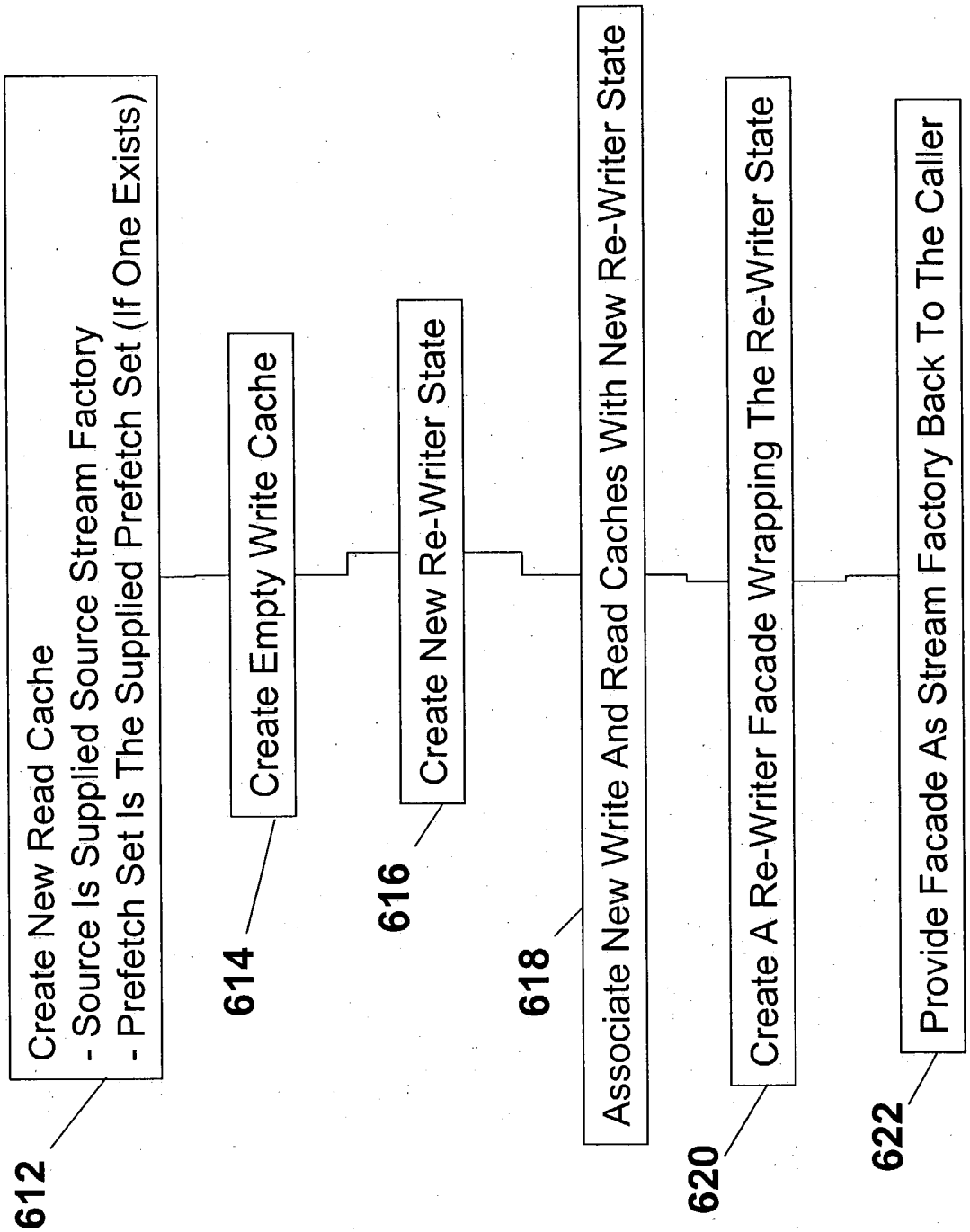

Create New Read Cache
- Source Is Supplied Source Stream Factory
- Prefetch Set Is The Supplied Prefetch Set (If One Exists)
612
Create Empty Write Cache
614
Create New Re-Writer State
616
Associate New Write And Read Caches With New Re-Writer State
618
Create A Re-Writer Facade Wrapping The Re-Writer State
620
Provide Facade As Stream Factory Back To The Caller
622

FIGURE 6

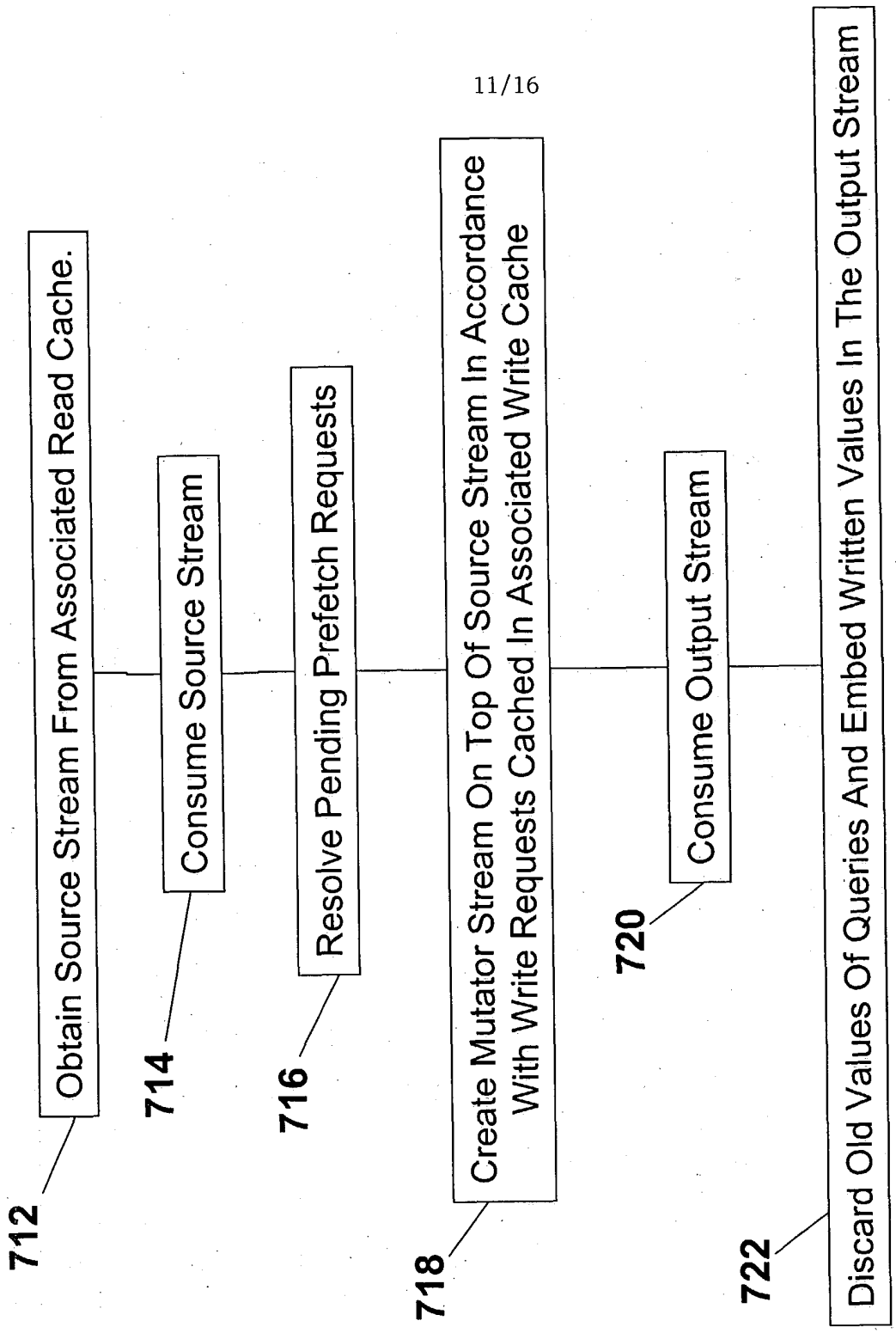

Obtain Source Stream From Associated Read Cache.
712
Consume Source Stream
714
Resolve Pending Prefetch Requests
716
Create Mutator Stream On Top Of Source Stream In Accordance With Write Requests Cached In Associated Write Cache
718
Consume Output Stream
720
Discard Old Values Of Queries And Embed Written Values In The Output Stream
722

FIGURE 7

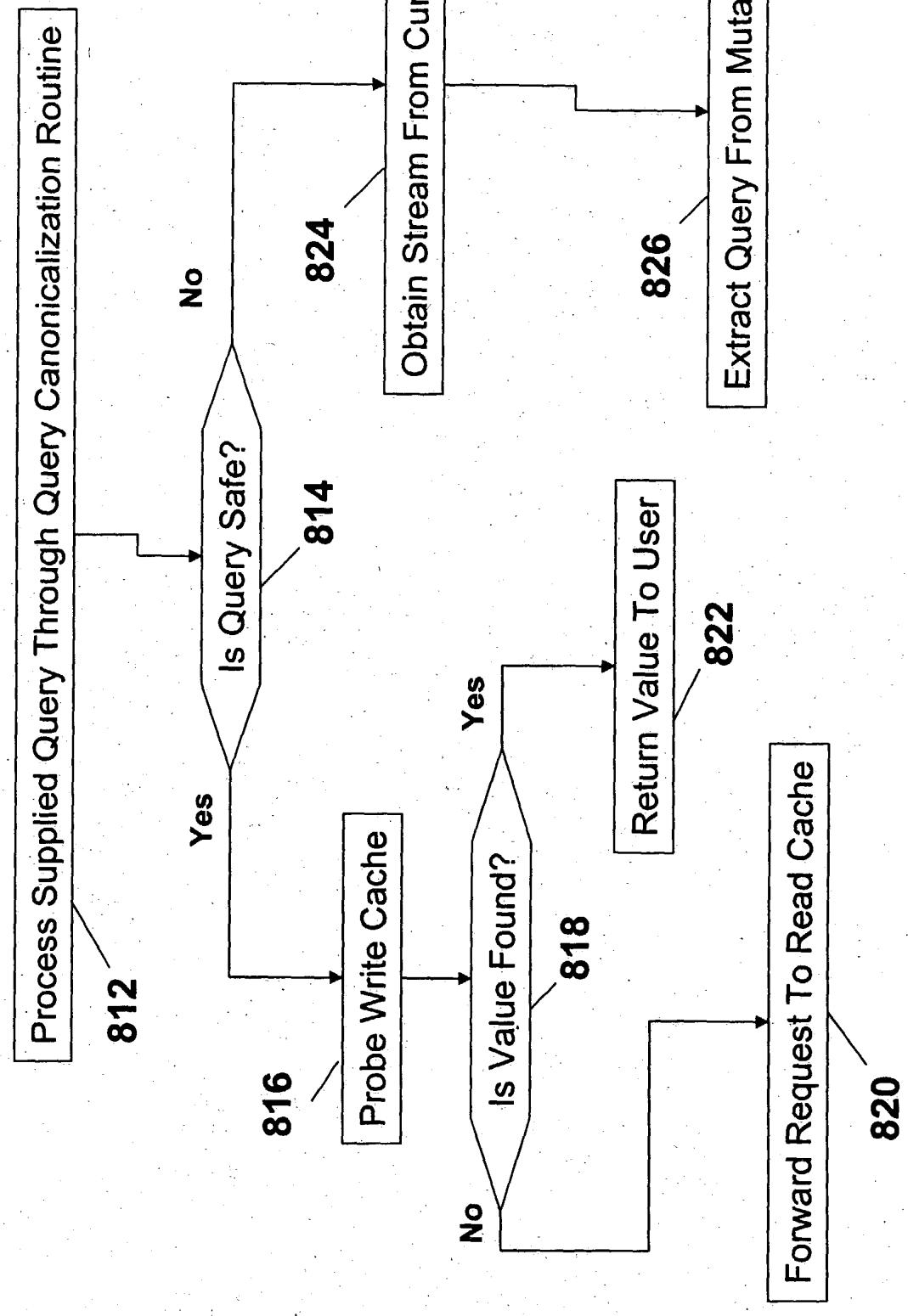

Process Supplied Query Through Query Canonicalization Routine
812
Is Query Safe?
814
No
Yes
Obtain Stream From Current Instance
824
Extract Query From Mutator Stream
826
Probe Write Cache
816
Is Value Found?
818
Yes
No
Return Value To User
822
Forward Request To Read Cache
820

FIGURE 8

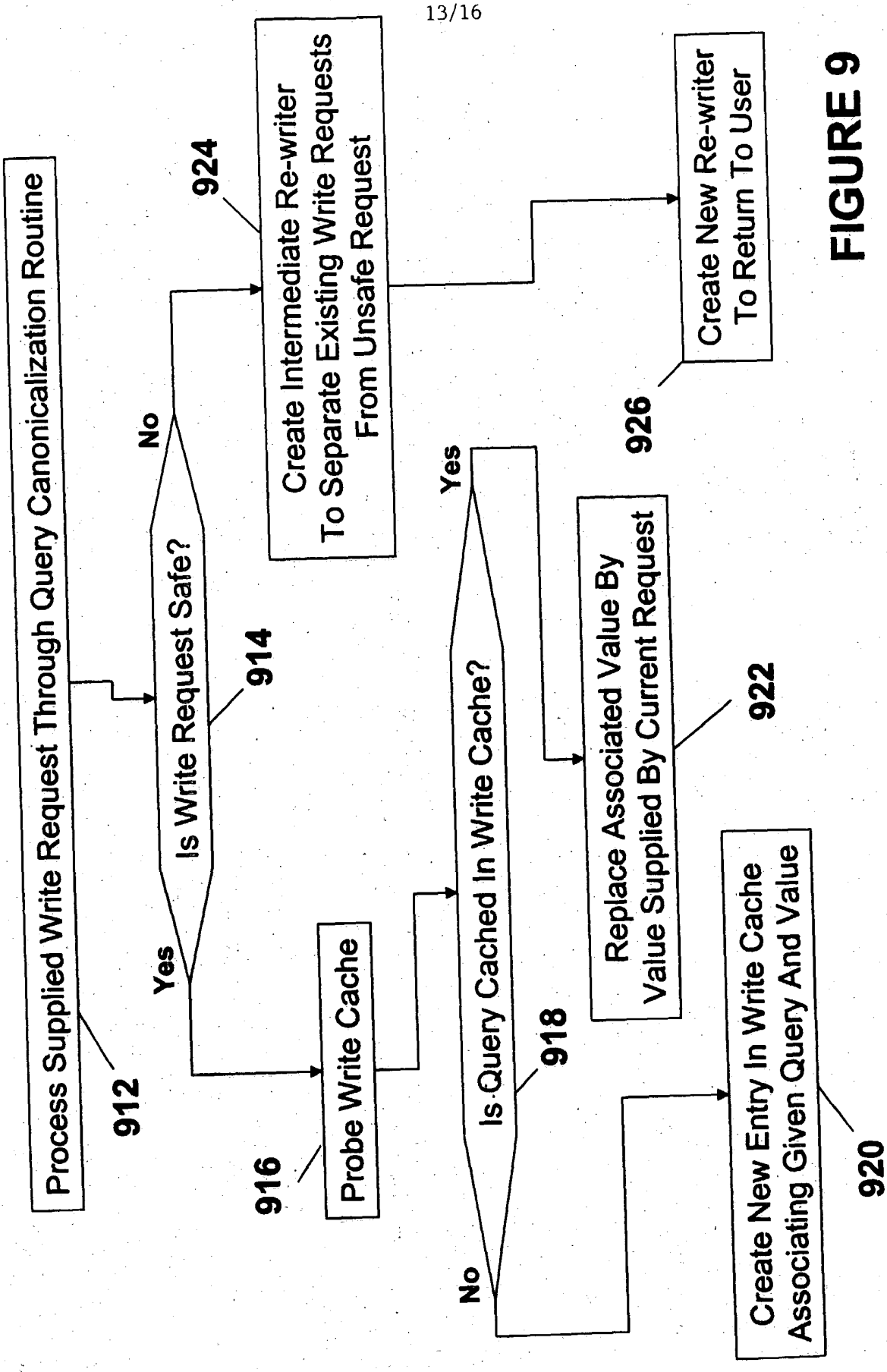

Process Supplied Write Request Through Query Canonicalization Routine
912
Is Write Request Safe?
914
No
924
Create Intermediate Re-writer To Separate Existing Write Requests From Unsafe Request
926
Create New Re-writer To Return To User
Yes
916
Probe Write Cache
Is Query Cached In Write Cache?
918
Yes
922
Replace Associated Value By Value Supplied By Current Request
No
920
Create New Entry In Write Cache Associating Given Query And Value

FIGURE 9

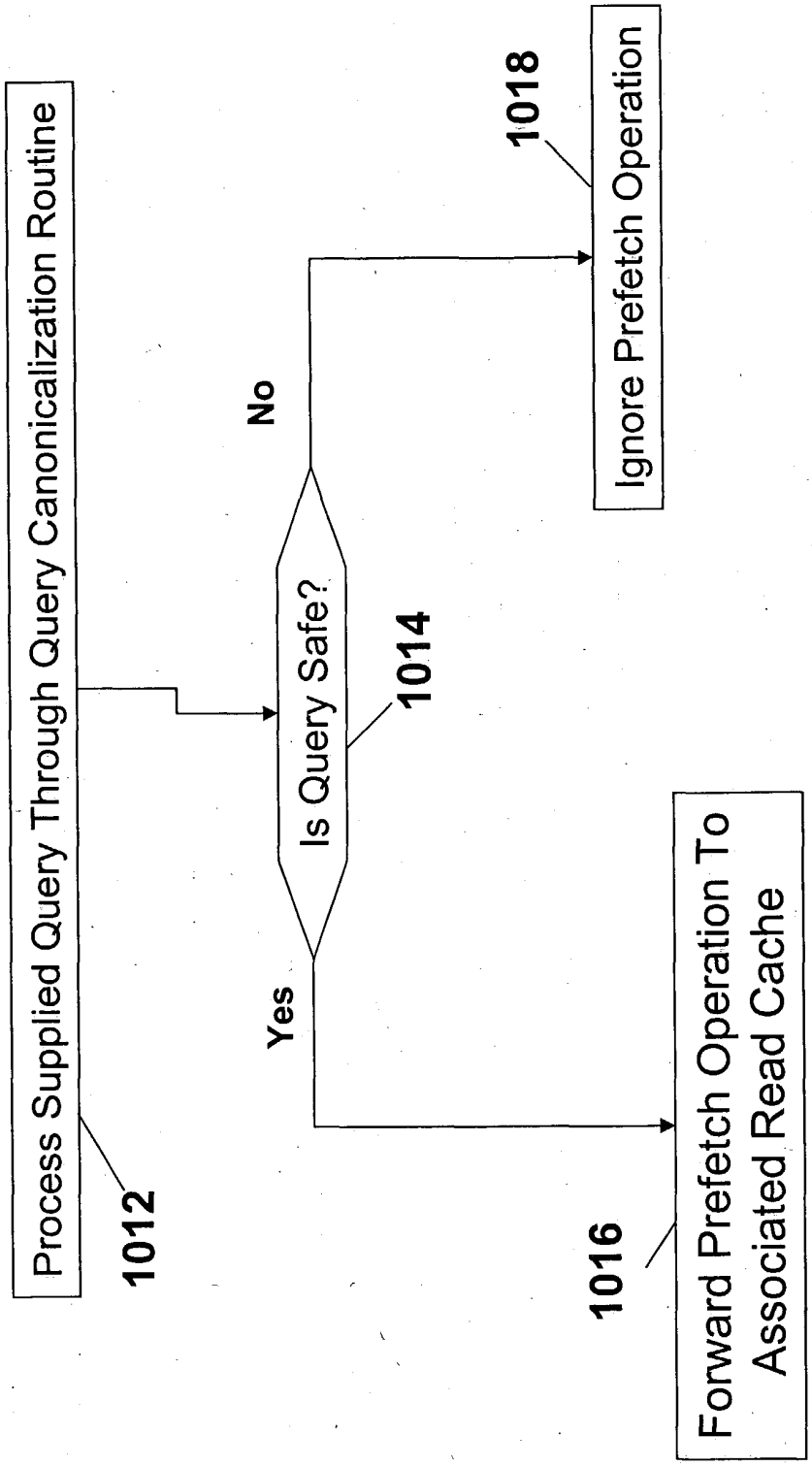

Process Supplied Query Through Query Canonicalization Routine
1012
Is Query Safe?
1014
No
Yes
Ignore Prefetch Operation
1018
Forward Prefetch Operation To Associated Read Cache
1016

FIGURE 10

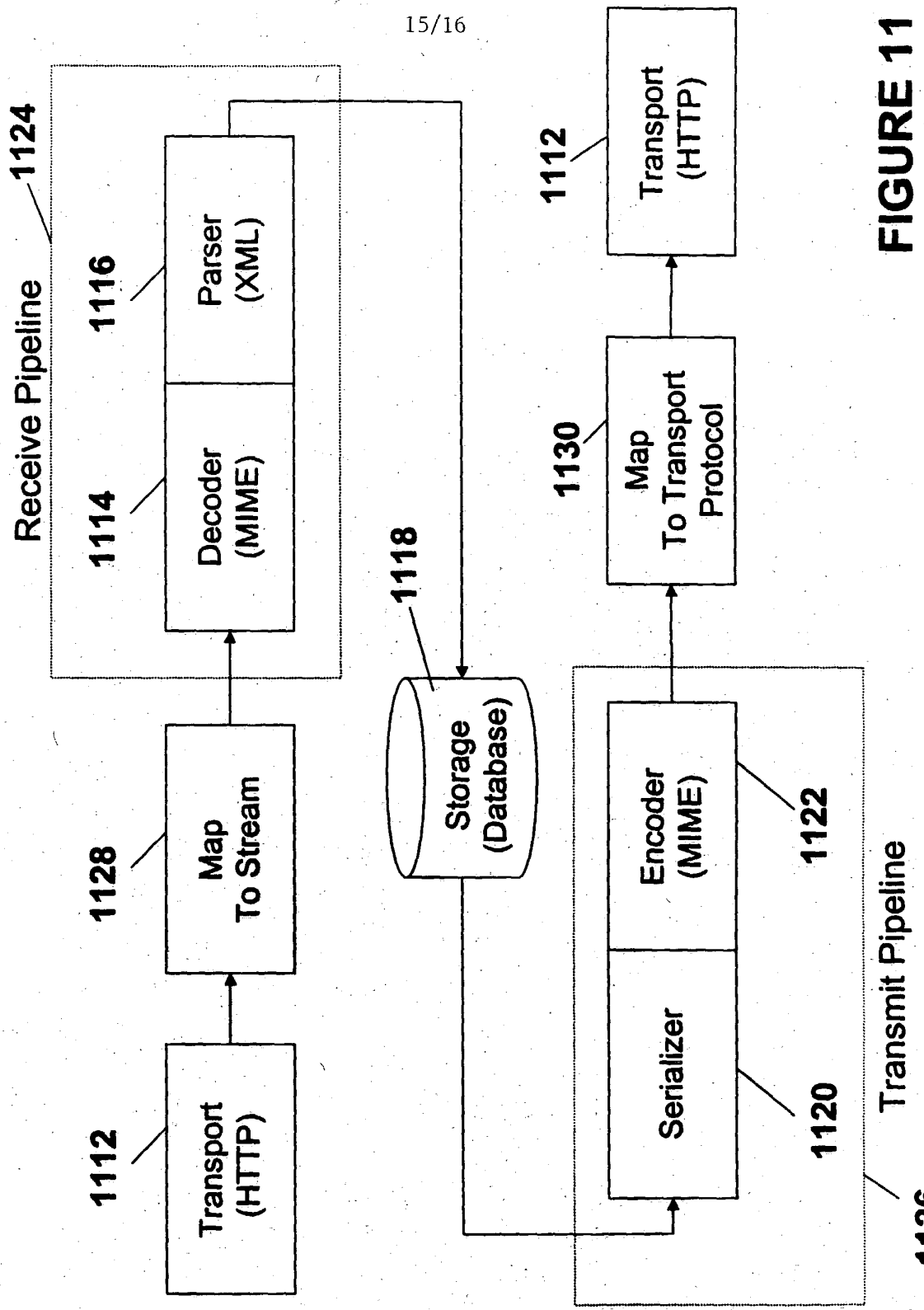

Receive Pipeline
1124
1116
Parser (XML)
1114
Decoder (MIME)
1128
Map To Stream
1112
Transport (HTTP)
1118
Storage (Database)
1112
Transport (HTTP)
1130
Map To Transport Protocol
1122
Encoder (MIME)
1120
Serializer
Transmit Pipeline
1126

FIGURE 11

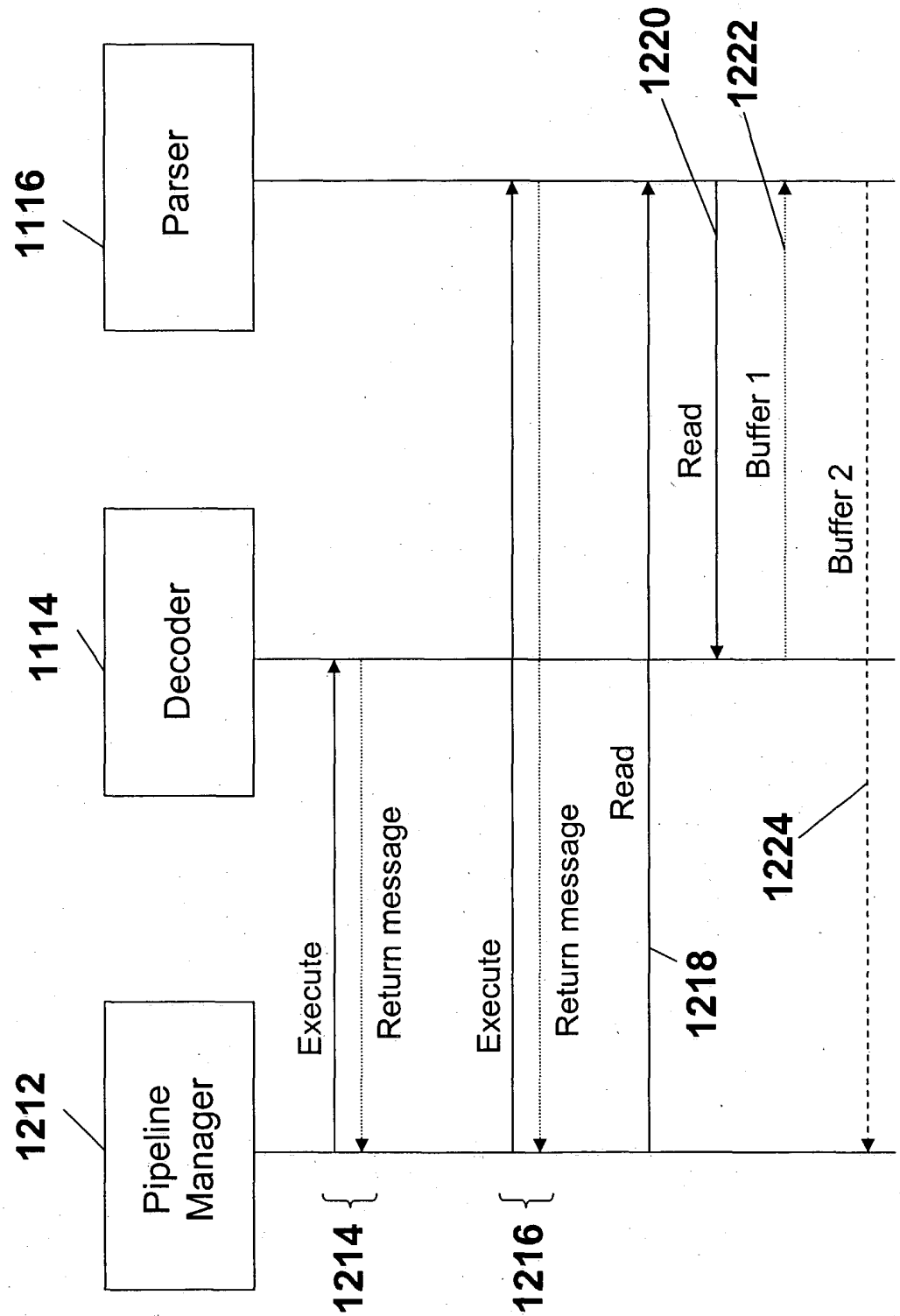

1116
Parser
1114
Decoder
1212
Pipeline Manager
1220
1222
Read
Buffer 1
Buffer 2
Execute
Return message
Execute
Return message
Read
1218
1224
1214
1216

FIGURE 12